sciendo

TATRA MOUNTAINS
Mathematical Publications

DOI: 10.2478/tmmp-2023-0032
Tatra Mt. Math. Publ. **87** (2024), 25–42

CRYPTANALYSIS OF A SPECIAL POLYBIUS-LIKE CIPHER USING HILL-CLIMBING

Eugen Antal

Institute of Computer Science and Mathematics,
Slovak University of Technology in Bratislava,
SLOVAKIA

ABSTRACT. Polybius cipher is a special substitution system widely used during history. In our research, we found several Polybius-like ciphers used in Czechoslovakia and in the Slovak State from the first half of the 20th century. Various types of this cipher are described in the first Czechoslovak cryptanalysis manual "šifrovací systémy a návod k luštění kryptogramů" by plk. Josef Růžek. It can be also found in the official cryptology directive called G–VII–8 "šifrování" (encryption) from 1938, and another variant in the new version of the same directive from 1946. In this work, we focus on a special Polybius-like cipher inspired by three real ciphers used in Czechoslovakia and in the Slovak State. Two were used during WW2 and one right after the war. We will show how these ciphers were used and how they can be solved with a modern heuristic approach on a personal computer. We evaluate the effectiveness of the Hill-Climbing heuristic methods with restarts. We also investigated several different fitness functions and language models.

1. Introduction

Polybius (c. 200 – c. 118 BC) was a Greek historian and writer who devised a special signaling system that was later adopted as a cryptographic method [7,14]. His system is called *Polybius square* (also called Polybius checkerboard [7]) and consists of a 5×5 square (table) filled with the letters of the alphabet. In the case of the English alphabet, the letters i and j are merged to fit the square. In the original version from Polybius, the rows and columns were labeled

© 2023 Mathematical Institute, Slovak Academy of Sciences.
2020 Mathematics Subject Classification: 68T20, 68T50.
Keywords: Military history, Slovakia, Czechoslovakia, Historical ciphers, Polybius cipher, Polybius checkerboard, Cryptanalysis, Hill-Climbing, Language models.
Supported by the grant VEGA 2/0072/20.
Licensed under the Creative Commons BY-NC-ND 4.0 International Public License.

with numbers $1, 2, 3, 4, 5$. These numbers serve as a row and column coordinate in the table. The encryption process transforms the letters in the square into their coordinates.

Encryption systems with similar characteristics have been widely used throughout history [7]. Polybius-like cipher from the seventeenth and eighteenth century can be found in the Hessian State Archives in Marburg, Germany [15]. In the twentieth century, countries like Slovakia [16,17] and former Czechoslovakia [18] (also many other countries) used similar ciphers.

In the following sections, we provide more details about the Polybius checkerboard cipher. We start with a brief summary of different cipher variants. We also provide some background on real cipher instances used during WW2 and after the war. We continue with the description of methods of how this cipher type can be transformed into an optimization problem and solved with modern heuristic-cryptanalysis techniques on a personal computer. The design of a suitable fitness function (scoring function) is crucial. Thus, we investigated several different fitness functions and language models. In the final part of the article, we summarize our obtained results.

2. Preliminaries

In this paper, we use the following notation:

- P — plaintext;
- C — ciphertext;
- $\mathcal{A}_P$ — plaintext alphabet;
- $\mathcal{A}_C$ — ciphertext alphabet;
- k — key (secret parameter of the cryptosystem);
- e_k — encryption function parameterized with a key;
- d_k — decryption function parameterized with a key.

Throughout the paper, $P = x_0, x_1, x_2, \ldots$ will represent letters of a plaintext and $C = y_0, y_1, y_2, \ldots$ will represent units of a ciphertext. We consider only plaintexts consisting of lower-case letters of the English alphabet $\{a, \ldots, z\}$, digits $\{0, \ldots, 9\}$ and special characters {comma, dot, question mark, dash }. Because some variants of the cipher can encrypt directly n-grams and words, the spaces between words are preserved in the input text but are not encrypted. The ciphertexts can only consist of lower-case letters of the English alphabet or their numerical representation[1]. In general, $\mathcal{A}_C$ can contain also special symbols, but for simplicity, we will restrict the $\mathcal{A}_C$ to letters and numbers only. We will consider an adversary Eve who has access to a ciphertext but does not know the original plaintext nor the key used during the encryption process.

[1]We can directly represent letters of the English alphabet by elements of $\mathbb{Z}_{26}$, 0 representing the letter a, 1 representing the letter b, and so on.

Moreover, Eve has additional information that the used plaintexts are meaningful texts from a natural language, and she knows the used language.

2.1. Polybius cipher variants

In the original version from Polybius, the rows and columns were labeled with numbers $1, 2, 3, 4, 5$ (see Table 1). These numbers serve as row and column coordinates in the table. The encryption process transforms the letters in the square to their coordinates (first we use the row coordinate, then the column coordinate). Each $x_i \in \mathcal{A}_P$ is encrypted with two elements of $\mathcal{A}_C$, thus the encryption function e_k is a bijective mapping $\mathcal{A}_P \to \mathcal{A}_C^2$. This version of the Polybius cipher is a simple (monoalphabetic) substitution cipher, where each letter is encrypted with a two-digit number. Figure 1 contains nine examples of the "classical"[2] Polybius cipher from the 17th–18th century.

Table 1. Polybius checkerboard.

	1	**2**	**3**	**4**	**5**
1	a	b	c	d	e
2	f	g	h	i/j	k
3	l	m	n	o	p
4	q	r	s	t	u
5	v	w	x	y	z

As an example, we can encrypt the *conference* plaintext using Table 1 as $c \to 13$, $o \to 34$, $n \to 33$, $f \to 21$, ... receiving the *13343321154215331315* ciphertext. The decryption process is straightforward. Because of the bijection, this method has a major drawback, the same letters are always encrypted to the same coordinates. Moreover, n-grams (neighbouring letters) produce also the same ciphertext sequence. Because of this weakness, several different variants of the cipher were created throughout history.

A popular way of improving its security was to increase the table size and include some frequently used letters of $\mathcal{A}_P$ multiple times, creating a homophonic substitution cipher. Alternatively, this can be also done in a different way, by adding multiple row/column coordinates for a specific table cell as shown in Figures 2 and 3. If we encrypt the same (conference) plaintext using the cipher key from Figure 2 (cipher key on the left), we obtain two possibilities for each of the (moderately frequented) letters c (e.g., $\mathrm{c} \to 29$, $\mathrm{c} \to 39$), o, f, and

[2]These Polybius cipher examples have only 20 elements (not 25 as in one designed by Polybius himself) and use letters as coordinates. The tables consist of 4 columns and 5 rows (or 5 columns and 4 rows).

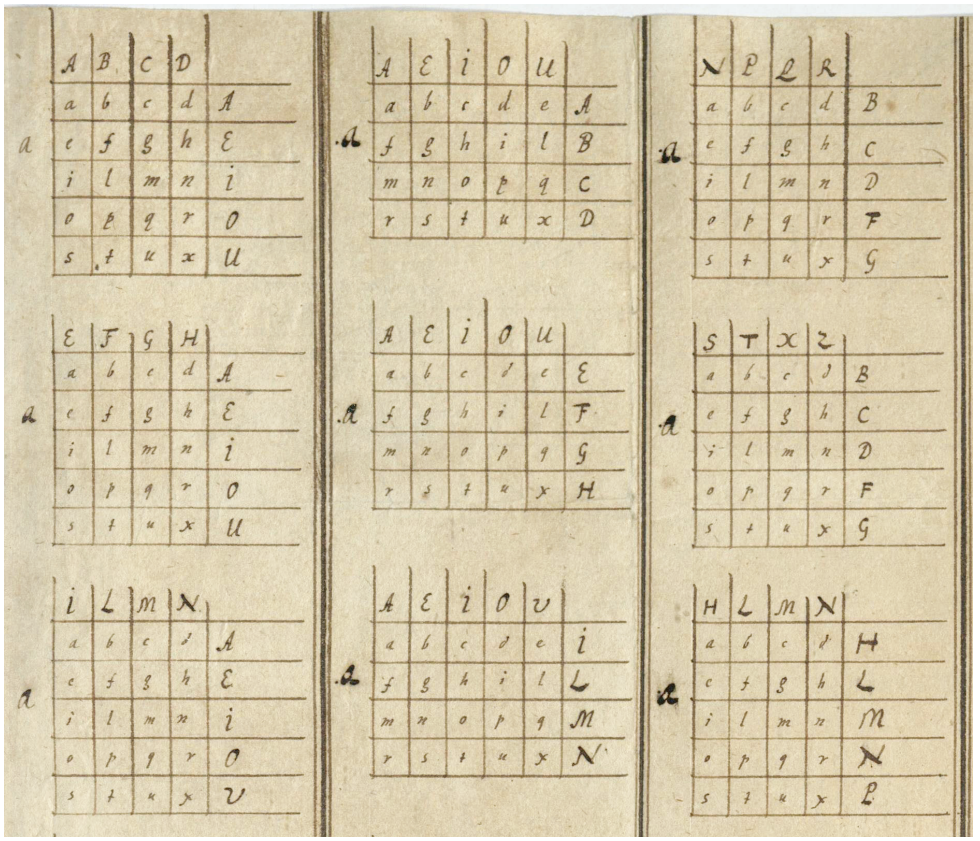


FIGURE 1. Polybius-like cipher examples from the 17th–18th century– –HStAM archive (HStAM 4d, Nr 1219).

six possibilities for the (most frequent) letters n, e, r.[3] We can now create several different variations of ciphertext for the same plaintext. However, it is still required to use the row/column coordinates in a specific order (row-column or column-row) because the labels are repeating. A possible option to bypass the need to use row-column order when reading coordinates is to use only unique labels. If we use the cipher key from Figure 2 (cipher key on the right), the CT coordinate (row-column reading order) stands for the same letter E as the TC (column-row reading order) coordinate. This trick can simplify the encryption process. In the case of the second example from Figure 3, every letter from $\mathcal{A}_P$

[3]So the first three letters of the plaintext ("con") can be encrypted in 24 (2x2x6) different possible ways.

can be encrypted in 25 different ways. In this case, the produced cipher text has good statistical properties (when the key is used properly[4]) that make it difficult to decipher.

A second way of improving the Polybius cipher was to extend the table content by adding numbers, common n-grams, letters with accents, special symbols, and common words (see Figure 4). Increasing the table content with common words and n-grams can also produce a cipher text with better statistical properties (when the key is used properly[5]).

In general, the Polybius checkerboard type cipher can vary in different aspects such as the table labels (numbers, letters, ...), table dimension (any rectangular shape not only a square), and table content (it can contain not only letters but also numbers, common bigrams, and words). It can also contain an element multiple times (homophonic substitution). A carefully designed cipher was considered a secure way of encryption. This is probably the reason why it can be found in many Slovak and Czechoslovak archival fonds[6]. Moreover, a Polybius-like cipher called "XY" was also included in the official cryptology directive called G-VII-8 named "šifrování"[7] (encryption) from 1938.[8]

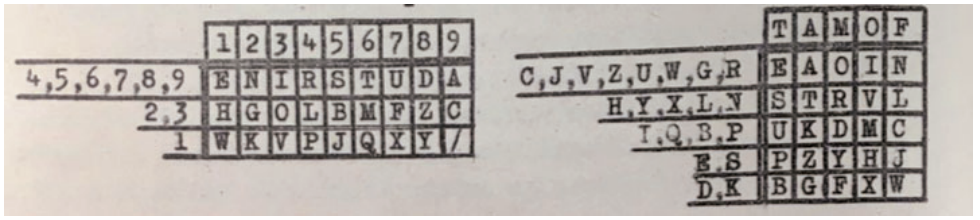


FIGURE 2. Polybius-like cipher examples from Czechoslovakia—Security Services Archive (Fonds of the Intelligence Department of the General Headquarters of the Czechoslovak Army, carton BF 389, part of file 027–19/7–0109, no. 389/1, "Šifrovací systémy a návod k luštění kryptogramů, I. díl, 1949").

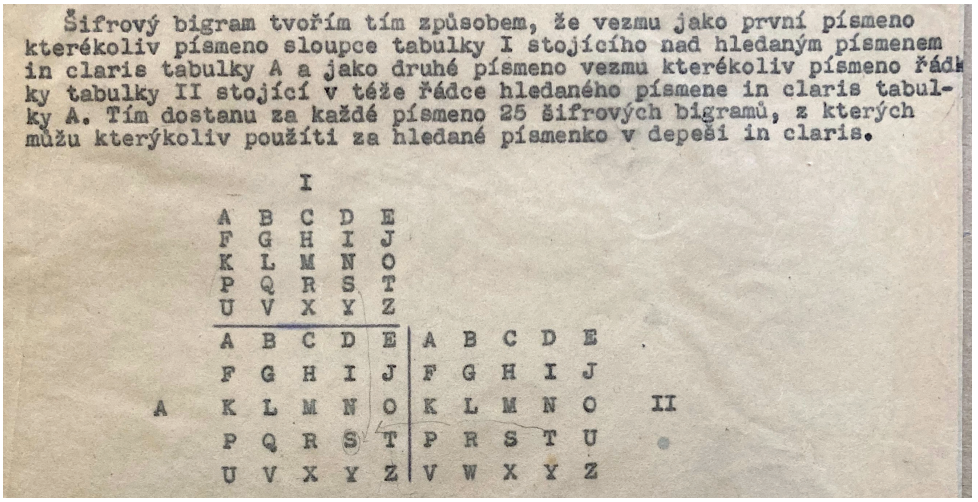
[4]Regular alternation of 25 coordinates for every letter.

[5]If we have the ability to encrypt whole words that are listed in the table, but the user of the cipher key it is not forced to actually do it, the security of the cipher may remain as in the case of an unextended table.

[6]The author found more Polybius-like ciphers used by Czechoslovakia than those included in this article.

[7]Another variant (a slightly different version) called "BITVA" was used in the new version of the same directive from 1946.

[8]The author plans to present a description of the encryption directive, including a description of the ciphers, in a future publication.


Šifrový bigram tvořím tím způsobem, že vezmu jako první písmeno
kterékoliv písmeno sloupce tabulky I stojícího nad hledaným písmenem
in claris tabulky A a jako druhé písmeno vezmu kterékoliv písmeno řádk
ky tabulky II stojící v téže řádce hledaného písmene in claris tabul-
ky A. Tím dostanu za každé písmeno 25 šifrových bigramů, z kterých
můžu kterýkoliv použíti za hledané písmenko v depeši in claris.

I

A	B	C	D	E	
F	G	H	I	J	
K	L	M	N	O	
P	Q	R	S	T	
U	V	X	Y	Z	

	A	B	C	D	E	A	B	C	D	E
	F	G	H	I	J	F	G	H	I	J
A	K	L	M	N	O	K	L	M	N	O
	P	Q	R	S	T	P	R	S	T	U
	U	V	X	Y	Z	V	W	X	Y	Z

II

FIGURE 3. Polybius-like cipher examples used after 1945 in Czechoslovakia—Security Services Archive (Fonds of the Intelligence Department of the General Headquarters of the Czechoslovak Army, carton BF 377, part of file 27–19/20–05, no. 5/81 "Šifr. klíč 'A–I/II' ").



	0	1	2	3	4	5	6	7	8	9
1	a	d	h	k	n	o	r	t	ů	x
2	á	e	ch	l	nn	p	s	tt	v	y
3	b	f	i	m	o	q	ss	u	w	z
4	c	g	j	mm	.	,	:	-	/	?
5	1	2	3	4	5	6	7	8	9	0

FIGURE 4. Polybius-like cipher examples from Czechoslovakia—Security Services Archive (Fonds of the Intelligence Department of the General Headquarters of the Czechoslovak Army, carton BF 389, part of file 027–19/7–0109, no. 389/1, "Šifrovací systémy a návod k luštění kryptogramů, I. díl, 1949").

3. Generalization of the Polybius cipher from real instances

In this work, we focus on a special Polybius-like checkerboard cipher inspired by three real ciphers used in Czechoslovakia and in the Slovak State. Two were used during WW2 and one right after the war. The first investigated Polybius-like checkerboard cipher is a 9×9 table. The table contains letters, numbers, and common bigrams/trigrams (from the Slovak language). Numbers $1, 2, \ldots, 9$ were used as row/column labels. It was used during army training in the Slovak State in 1940 [17]. This cipher was called "šifrovací kľúč" (cipher key). The row and column labels were permanent, but the order of the columns was changed daily based on an additional permutation. Fortunately, the whole instruction of use was preserved in the archive (Figure 5). The first step during the encryption was to split the plaintext into syllables and to check if the cipher key contains them. If a syllable was present in the table[9] it was encrypted, otherwise, it was split into letters and then encrypted letter-wise.

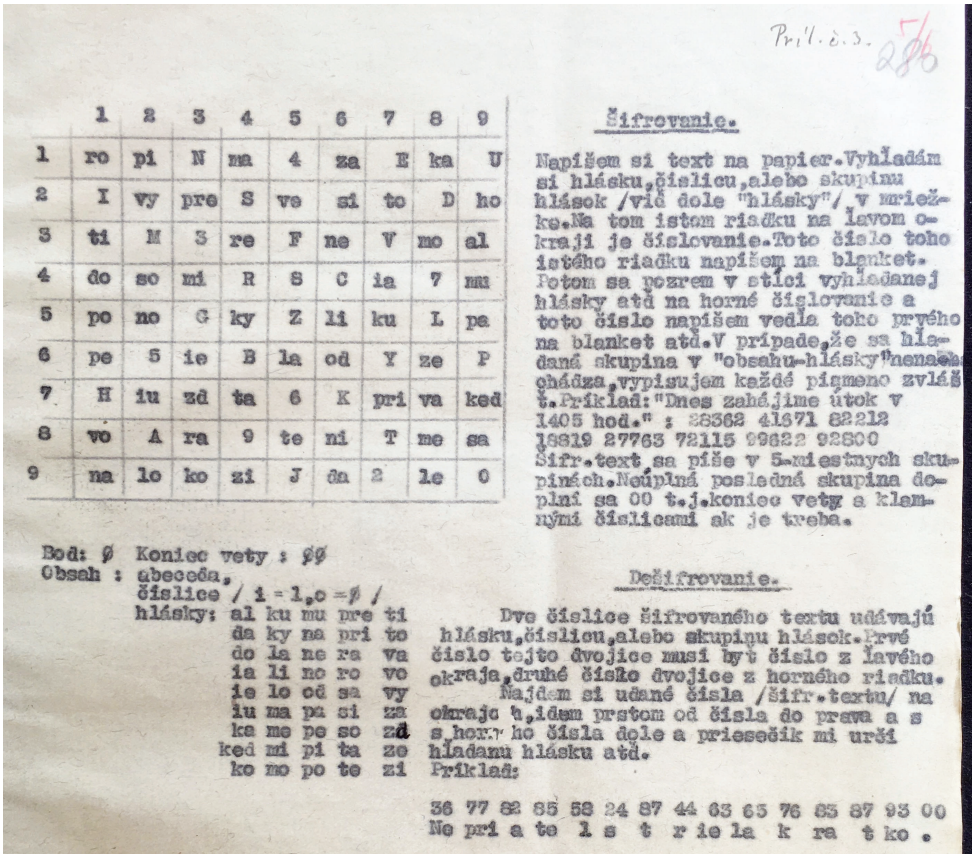
In 1944, a very similar cipher (Figure 6) was used by the Czechoslovak resistance army fighting on the Eastern front called the 1st Czechoslovak Army Corps in the Soviet Union [18]. The table size was 10×10 (labels $0, 1, \ldots, 9$). It contains letters, numbers, common bigrams/trigrams (from the Czech language), and special characters (e.g., dot, comma, ...). Some letters and bigrams are with a diacritical mark. This cipher was called *encryption table No. 58 C* and was used in radio communication. Both the row and column labels were changed daily.

The third Polybius-like checkerboard cipher (Figure 7) was used in the newly established Czechoslovakia in 1947 by army units fighting against the Ukrainian nationalist paramilitary and partisan formations. The table size, in this case, was 10×10 and it contains uppercase and lowercase letters, numbers, common bigrams, words, and special characters. This variant also contains elements with a diacritical mark. The column labels were permanent and the row labels were set based on the key.

4. Modern cryptanalysis of classical ciphers

A very popular method of solving historical ciphers is using optimization algorithms (meta-heuristics). This problem has been well studied in recent years.

[9]There was attached an additional list of used syllables.



	1	2	3	4	5	6	7	8	9
1	ro	pi	N	ma	4	za	E	ka	U
2	I	vy	pre	S	ve	si	to	D	ho
3	ti	M	3	re	F	ne	V	mo	al
4	do	so	mi	R	S	C	ia	7	mu
5	po	no	G	ky	Z	li	ku	L	pa
6	pe	5	ie	B	la	od	Y	ze	P
7	H	iu	zd	ta	6	K	pri	va	ked
8	vo	A	ra	9	te	ni	T	me	sa
9	na	lo	ko	zi	J	da	2	le	0

Šifrovanie.

Napíšem si text na papier.Vyhľadám si hlásku,číslicu,alebo skupinu hlások /vid dole "hlásky"/ v mriežke.Na tom istom riadku na ľavom okraji je číslovanie.Toto číslo toho istého riadku napíšem na blanket. Potom sa pozrem v stĺpci vyhľadanej hlásky atď na horné číslovanie a toto číslo napíšem vedľa toho prvého na blanket atď.V prípade,že sa hľadaná skupina v "obsahu-hlásky"nenachádza,vypisujem každé písmeno zvlášť.Príklad:"Dnes zahájime útok v 1405 hod." : 28362 41671 82212 19819 27763 72115 09822 92800 Šifr.text sa píše v 5-miestnych skupinách.Neúplná posledná skupina doplní sa 00 t.j.koniec vety a kľúčovými číslicami ak je treba.

Bod: ø Koniec vety : øø
Obsah : abeceda,
číslice / i = 1,o =ø /
hlásky: al ku mu pre ti
da ky na pri to
do la ne ra va
ia li no ro vo
ie lo od sa vy
iu ma pa si za
ka me pe so zd
ked mi pi ta ze
ko mo po te zi

Dešifrovanie.

Dve číslice šifrovaného textu udávajú hlásku,číslicu,alebo skupinu hlások.Prvé číslo tejto dvojice musí byť číslo z ľavého okraja,druhé číslo dvojice z horného riadku. Nájdem si udané čísla /šifr.textu/ na okraji a,idem prstom od čísla do prava a s s hor.ho čísla dole a priesečník mi určí hľadanú hlásku atď.
Príklad:

36 77 82 85 58 24 87 44 63 65 76 85 87 93 00
Ne pri a te l s t r ie la k ra t ko .

FIGURE 5. VHA Bratislava, f. MNO d., box 34.

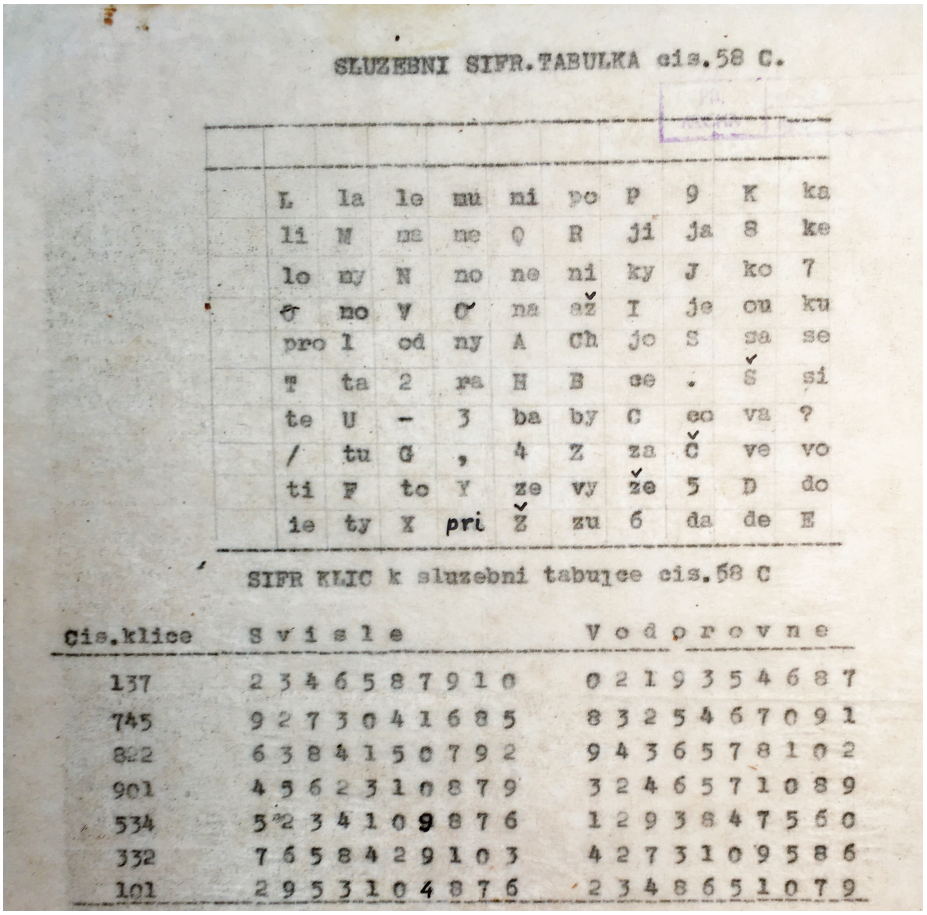
For more details we recommend the following publications to the reader: [1–3, 5, 9–11, 13]. We briefly summarize the main idea in the rest of this section.

The main principle of a meta-heuristic method (e.g., Hill-Climbing, Genetic Algorithm, etc.) is an iterative search process. First, we generate solution candidates, and then we try to improve their quality through several computation cycles by changing them. In most cases, the initial solution is randomly generated. In each iteration, a changing process is performed on these solution candidates and their quality is calculated. The quality of a solution candidate is evaluated using a fitness (scoring) function. The goal of an optimization

algorithm (meta-heuristic method) is to reach the global optimum of the fitness function. The pseudo-code of the Hill-Climbing method is in Algorithm 1.

Algorithm 1 Hill-Climbing algorithm

1: Generate an initial random solution candidate S; iteration counter $k = 0$;
2: Find the neighbours $N(S)$ of S;
3: Randomly pick an element S' from $N(S)$.
4: Accept S' as a new solution candidate $S = S'$, if S' has better fitness than S.
5: Next iteration $k \leftarrow k + 1$.
6: While the termination criteria are not satisfied, jump to step 2.



SLUZEBNI SIFR.TABULKA cis.58 C.

L	la	le	mu	ni	po	p	9	K	ka
li	M	me	ne	Q	R	ji	ja	8	ke
lo	my	N	no	ne	ni	ky	J	ko	7
O	no	V	O	na	až	I	je	ou	ku
pro	I	od	ny	A	Ch	jo	S	sa	se
T	ta	2	ra	H	B	se	.	š	si
te	U	-	3	ba	by	C	eo	va	?
/	tu	G	,	4	Z	za	č	ve	vo
ti	F	to	Y	ze	vy	že	5	D	do
ie	ty	X	pri	ž	zu	6	da	de	E

SIFR KLIC k sluzebni tabulce cis.58 C

Cis.klice	S v i s l e	V o d o r o v n e
137	2 3 4 6 5 8 7 9 1 0	0 2 1 9 3 5 4 6 8 7
745	9 2 7 3 0 4 1 6 8 5	8 3 2 5 4 6 7 0 9 1
822	6 3 8 4 1 5 0 7 9 2	9 4 3 6 5 7 8 1 0 2
901	4 5 6 2 3 1 0 8 7 9	3 2 4 6 5 7 1 0 8 9
534	5 2 3 4 1 0 9 8 7 6	1 2 9 3 8 4 7 5 6 0
332	7 6 5 8 4 2 9 1 0 3	4 2 7 3 1 0 9 5 8 6
101	2 9 5 3 1 0 4 8 7 6	2 3 4 8 6 5 1 0 7 9

FIGURE 6. VÚA Praha, f. 1. čs. spoj. prapor v SSSR 1944–1945, box 2.

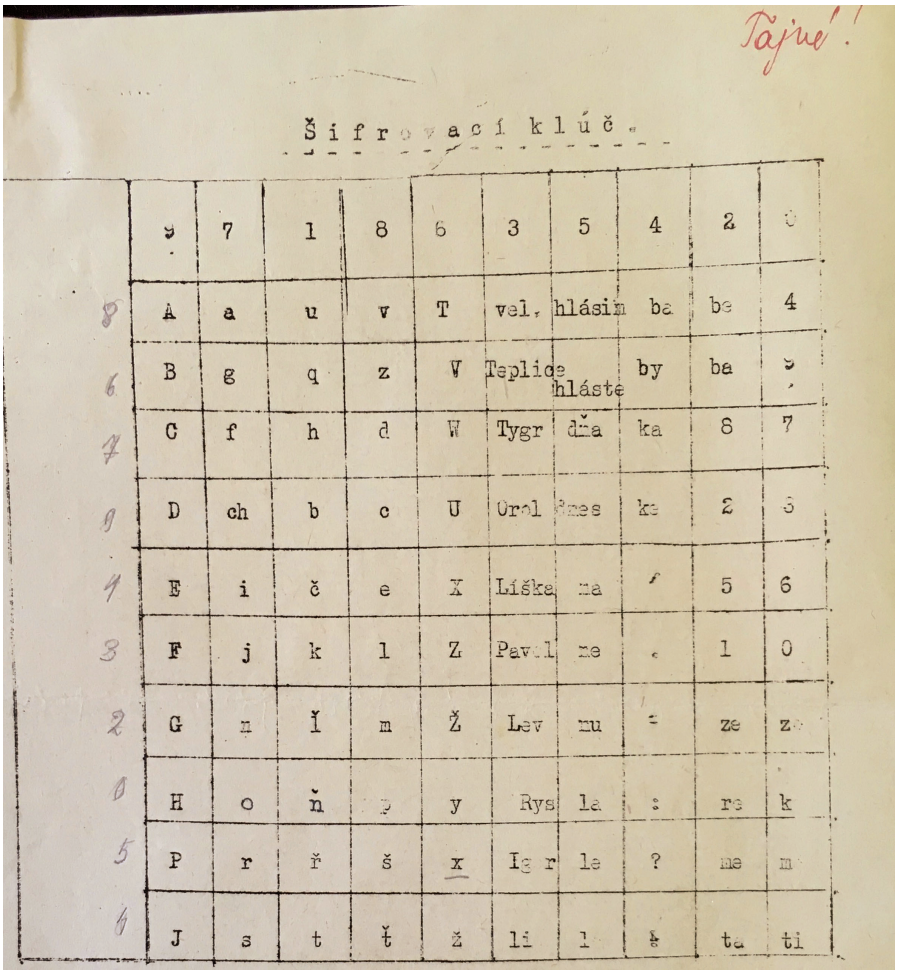


FIGURE 7. VHA Bratislava, f. Banderovci, box 77.

The idea behind the heuristic cryptanalysis is to transform the keyspace search into an optimization problem. The goal is to find the correct key of a selected cryptosystem. In many cases only the key is unknown. The main reason why the meta-heuristic methods work on most of the historical ciphers is that these ciphers have a low level of diffusion. It means that the partially correct key produces a partially correct plaintext. From this point of view, we need a fitness function that can calculate how much is the resulting plaintext candidate correct. If we assume that the correct result is a meaningful text, we can calculate this meaningfulness based on some statistical distance/similarity of the quantitative characteristic of the text elements (e.g., n-gram frequencies).

5. Experiments

We investigated the possibility of solving a special Polybius-like cipher (consisting of letters, numbers, bigrams/trigrams, and special symbols) with a modern heuristic approach. In the experiments, we used English texts of different lengths and English statistics in fitness functions. We used several English-written text files of lengths $\{200, 500, 1000, 2000\}$, respectively. For each text length, we tested 100 different texts.

The cryptanalysis (solving) task was defined as a keyspace search, where the daily key is the only unknown parameter. This task was transformed into an optimization problem. We evaluate the effectiveness (success rate) of the Hill-Climbing meta-heuristic methods with 100 restarts. For each text, we restarted the Hill-Climbing 100 times obtaining 100 results. Next, we calculated the *average match rate*[10] for each text length from 100×100 results. We also saved the best match rate for each text (the best result among the 100 restarts) and calculated the *average maximal match rate* for each text length from 100 results. The pseudo-code of the used Hill-Climbing instance in our experiments is in Algorithm 2.

In our experiments, we investigated several different fitness functions[11] (L1 distance, Jensen-Shannon divergence, weighted sum) and language models (n-gram for $n = \{1, 2, 3, 4\}$).

To evaluate how the problem complexity depends on the table dimension, we created several custom Polybius-like ciphers with different sizes from 5×5 up to 10×10. The smallest table contains only letters (it represents the classical Polybius cipher). By increasing the table size we extended the table by adding numbers, bigrams, trigrams, and special symbols. The 6×6 table contains letters and numbers only. Table sizes from 7×7 to 9×9 contain also bigrams in addition. Bigrams were added in an order based on their frequency in the English language. The 10×10 table contains, in addition, the five most frequent trigrams (the, and, ing, ion, ent) and some special symbols.

Our experiments were performed on a modern computer with the following configuration:

- Intel i9-7940X CPU (14 cores / 28 log. processors),
- 128 GB RAM,
- Samsung EVO SSD disk.

The total computation time on 28 threads was approx. 38 hours.

[10]The number of corresponding letters of a result with the correct plaintext converted to a percentage.

[11]For more details see [1] and [3].

Algorithm 2 Used Hill-Climbing method

Input: mi - max iterations ▷ termination criteria
Input: l - key length
Input: d_k - decryption function
Input: f - fitness (scoring) function
Input: C - cipher text
Output: Key candidate k with the best score

1: Generate an initial random key candidate k ▷ random permutation of the column/row labels of length l
2: Iteration counter $i = 0$
3: Calculate the score of k as $s_k = f(d_k(C, k))$ ▷ based on the n-gram statistics from the decrypted text
4: **while** $i < mi$ **do**
5: Generate new key candidate k_c from k by randomly swapping two elements in the row/column labels ▷ with equal probability the swap is performed only in the row label; only in the column label; or both, in row and column labels
6: Calculate the score of k_c as $s_{kc} = f(d_k(C, k_c))$
7: **if** s_{kc} is better than s_k **then**
8: $k \leftarrow k_c$
9: $s_k \leftarrow s_{kc}$
10: **end if**
11: $i = i + 1$
12: **end while**
13: Return k

To simulate the three real cipher types, we tested a variant where only the column labels were changed and a case where both column and row labels were parameters. In the first case (the solution space is small), we obtained a 100 % success rate (average maximal match rate) in almost all cases. Worse results were only in the case of smaller text sizes and large table sizes. For the second case, we were able to reach a 100 % success rate (average maximal match rate) for all text lengths and fitness functions only in the case of table size 5×5 (Figure 8), and 6×6. However, the average success rate for these table sizes was only 50 % and less (see Figure 9 for table size 5×5). It means that without restarts we are not able to solve the input texts.

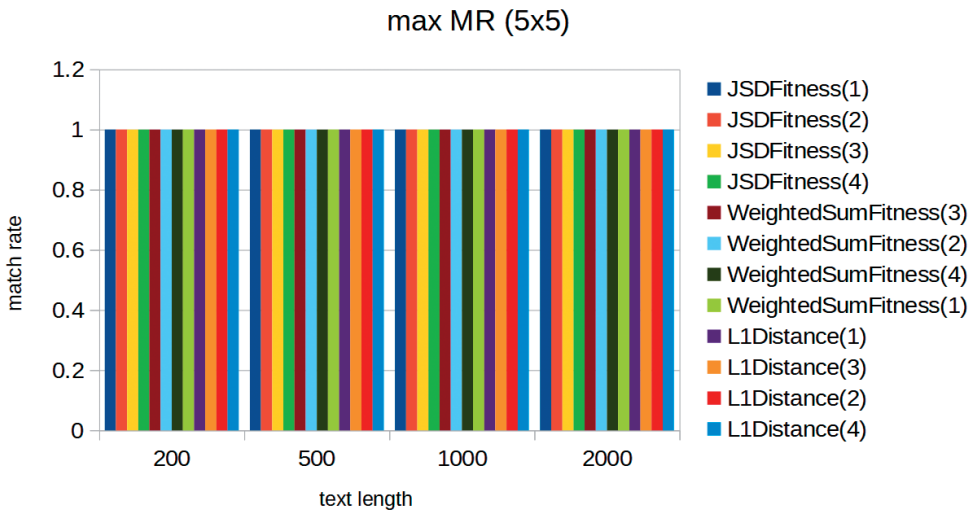


FIGURE 8. Average maximal match rate for table size 5×5.

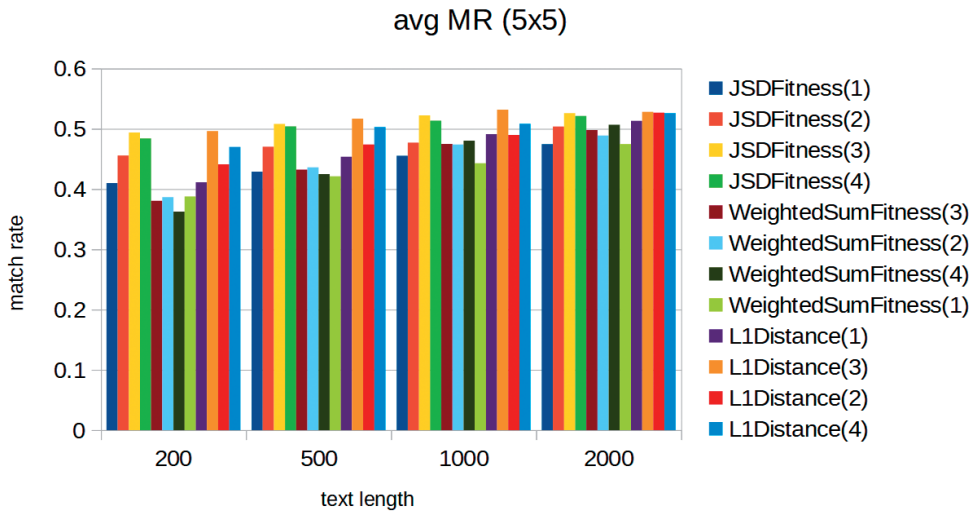


FIGURE 9. Average match rate for table size 5×5.

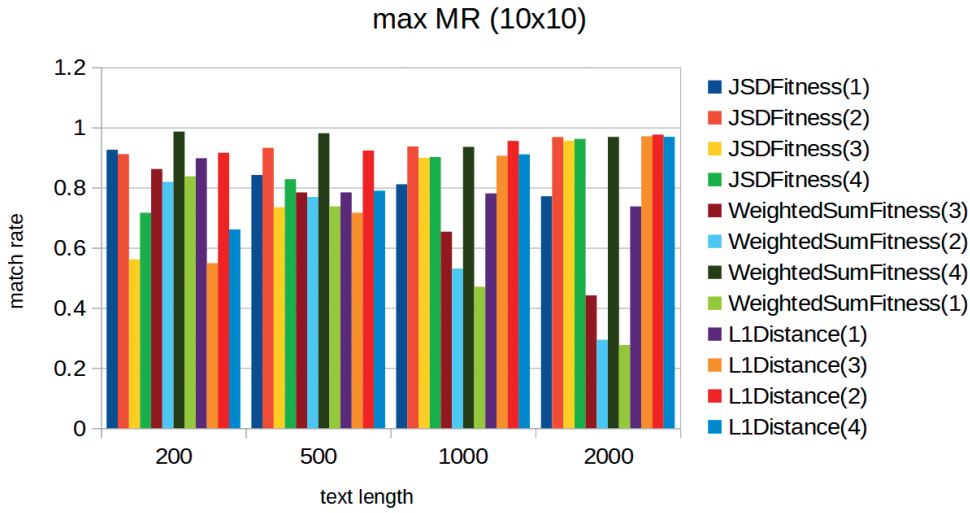


FIGURE 10. Average maximal match rate for table size 10×10.

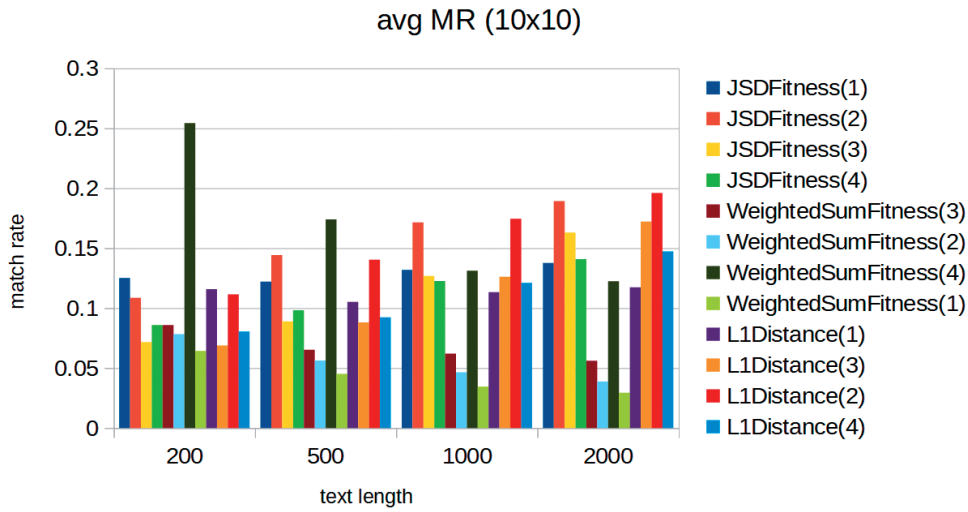


FIGURE 11. Average match rate for table size 10×10.

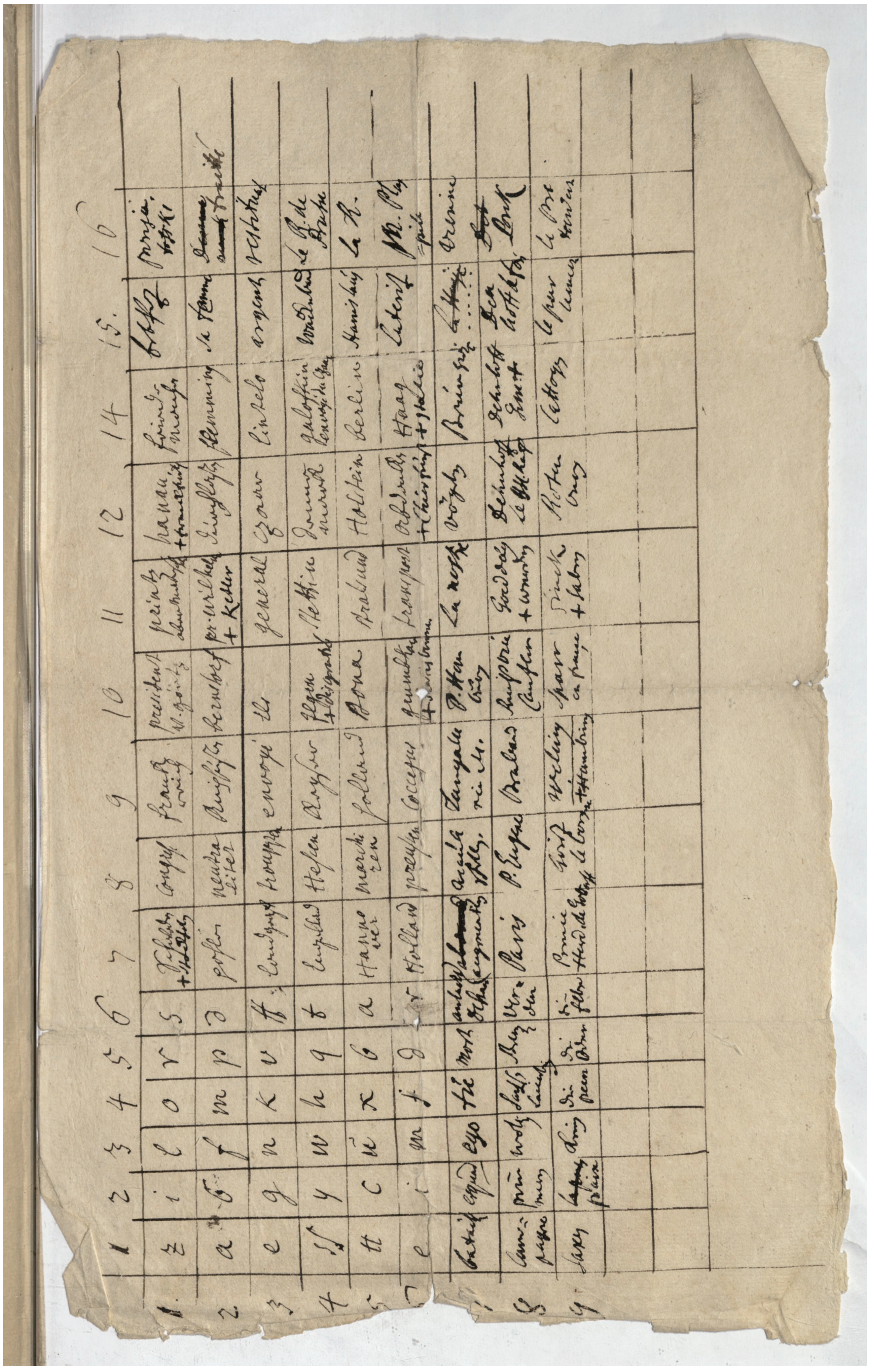


FIGURE 12. HStAM archive (HStAM, 4 f Schweden, nr. 253).

For table sizes > 6, the success rate varies according to the used language model and fitness function. We were able to solve all tested encrypted text files, but not for all fitness functions/language models. The obtained results (average match rate and average maximal match rate) for the largest table size 10×10 (the hardest problem instance) are in Figures 10 and 11.

The *average maximal match rate* is the most indicative quality measurement because it includes the added value of restarts. In general, the success rate varies according to the fitness function, language model, and text length. In our experiments, L1 distance and Jensen-Shannon divergence fitness functions performed the best.

6. Cipher Challenge

Solving a classical Polybius-like cipher is easy (even without a computer). However, the cipher's security can be improved in several ways, e.g., increasing the table size, adding table entries multiple times (creating a homophonic cipher), and including common n-grams and/or words. In general (if the cipher is not too complicated), solving such a modified Polybius-like cipher (similar instances that were used in the past) is still possible using modern computers in a short time. In this paper, we presented a well-working method for solving modified Polybius-like ciphers (of a given structure). This method has, however, some limitations. The efficiency of the method is decreasing with the increasing table size. Adding a large number of words (not only n-grams) to the key has also a negative impact on the success rate.

For this reason, we present a cipher challenge to the reader. In Figure 12 there is a real example of a Polybius-like cipher from Stockholm, from 1722. The preserved cipher key consists of 36 letters (homophonic cipher) and words. The aim is to solve an artificially created ciphertext of various lengths (200, 500, 1000, and 2000 letters long) encrypted by the presented cipher key.

During the encryption please use the following rules:

(1) Process the input text word by word.

(2) If the word is presented in the table (Figure 12), encrypt it with its coordinates (order: row and column), otherwise split the word into letters, and encrypt the letters with their coordinate (order: row and column) using the same table.

The proposed solving method should be based on a meta-heuristic method with a language model-based text evaluation function. Also, the artificial text must contain words from the key (at least half of the text must be covered

by the words from the key). Be aware — if the text doesn't contain any of the words from the key, it is only a classical version of a (homophonic) Polybius cipher. For simplicity, the row and column labels (of the cipher key) are the only unknown parameters in the solving process.

REFERENCES

[1] ANTAL, E.: *Modern Cryptanalysis of Classical Ciphers*. PhD. Thesis, STU in Bratislava, 2017. (In Slovak)

[2] ANTAL, E.—ELIÁŠ, M.: *Evolutionary computation in cryptanalysis of classical ciphers*, Tatra Mt. Math. Publ. **70** (2017), 179–197.

[3] ANTAL, E.—JAVORKA, P.—HLIBOKÝ, T.: *Cryptanalysis of the columnar transposition using meta-heuristics*. Tatra Mt. Math. Publ. **73** (2019), 39–60.

[4] BANKS, M. J.: *A Search-Based Tool for the Automated Cryptanalysis of Classical Ciphers*. Department of Computer Science, The University of York, May 2008.

[5] CLARK, A. J.: *Optimisation Heuristics for Cryptology*. PhD. Thesis, Queensland University of Technology, 1998.

[6] GANESAN, R.—SHERMAN, A.: *Statistical techniques for language recognition: An introduction and guide for cryptanalysts*, Cryptologia **17** (1993), no. 4, 321–366.

[7] KAHN, D.: *The Codebreakers*. Scribner, 1996.

[8] KULLBACK, S.: *Statistical Methods in Cryptanalysis*. Aegean Park Press, 1976.

[9] LASRY, G.: *A Methodology of Cryptanalysis of Classical Ciphers with Search Metaheuristics*. PhD. Thesis, Kassel University, 2017.

[10] MATTHEWS, R. A. J.: *The use of genetic algorithms in cryptanalysis*, Cryptologia **17** (1993), no. 2, 187–201.

[11] RUSSELL, M. D.—CLARK, J. A.—STEPNEY, S.: *Making the most of two heuristics: breaking transposition ciphers with ants*. In: CEC 03 Vol. 4, (2023), pp. 2653–2658.

[12] RŮŽEK, J.: *Šifrovací Systémy a Návod k Luštění Kryptogramů*. Praha, 1926–1949.

[13] SPILLMAN, R.—JANSSEN, M.—NELSON, B.: *Use of a genetic algorithm in the cryptanalysis of simple substitution ciphers*, Cryptologia **17**, (1993), no. 1, 31–44.

[14] VON ZUR GATHEN, J.: *CryptoSchool*. Springer-Verlag, Berlin, 2015.

[15] Hessisches Landesarchiv—Hessisches Staatsarchiv Marburg (HLA-HStAM).

[16] Vojenský Historický Archív, Bratislava, f. Banderovci.

[17] Vojenský Historický Archív, Bratislava, f. MNO dôverné.

[18] Vojenský Historický Archiv, Praha, f. 1. čs. spoj. prapor v SSSR, 1944–1945.

Received September 29, 2022
Revised September 21, 2023
Accepted September 22, 2023
Publ. online December 20, 2023

Institute of Computer Science and Mathematics
Slovak University of Technology in Bratislava
Ilkovičova 3
841 04 Bratislava
SLOVAKIA
E-mail: eugen.antal@stuba.sk